

**AMENDMENTS TO THE CLAIMS**

This listing of claims will replace all prior versions of claims in the application:

**Listing of Claims:**

1. (Currently Amended) A computer-implemented malware detection system for determining whether an executable script is malware according to its functionality, the malware detection system comprising:

a malware signature store including at least one known malware script signature, wherein each malware signature in the malware signature store is a normalized signature of a known malware script; and

a normalization module that obtains an executable script and generates a normalized signature for the executable script, wherein generating a normalized signature for the executable script comprises ~~translating~~ normalizing tokens from the executable script into normalized tokens conforming to a common format;

wherein the malware detection system is configured to:

compare the normalized signature of the executable script to the at least one normalized malware signature in the malware signature store to determine whether the executable script is malware; and

report whether the executable script is malware according to the determination.

2. (Previously Presented) The malware detection system of Claim 1, further comprising a comparison module, wherein the comparison module compares the normalized signature of the executable script to the at least one normalized malware signature in the malware signature store.

3. (Currently Amended) A computer-implemented malware detection system for determining whether an executable script is malware, the malware detection system comprising:

a malware signature storage means including at least one known malware signature, wherein each malware signature in the malware signature store means is a normalized signature

of a known malware script;

a normalization means that obtains an executable script and generates a normalized signature for the executable script, wherein the normalized signature for the executable script comprises a set of normalized tokens ~~translated~~ normalized from corresponding tokens in the executable script into a common format suitable for comparison with the at least one malware signature in the malware signature store means; and

a comparison means that compares the normalized signature for the executable script to the at least one malware signature in the malware signature storage means;

wherein the malware detection system is configured to determine whether the executable script is malware according to the comparison performed by the comparison means, and report whether the executable script is malware.

4. (Currently Amended) A computer-implemented method for determining whether a computer-executable script is malware, the method comprising[[:]]

using one or more processors to perform the following computer-executable acts:

obtaining an executable script;

generating a first normalized signature for the executable script, wherein the first normalized signature comprises normalized tokens ~~translated~~ normalized from corresponding tokens in the executable script in a format suitable for comparison to normalized signatures of known malware;

comparing the first normalized signature to at least one normalized signature of known malware;

determining, based on the previous comparison, whether the executable script is malware; and

reporting the results of the determination as to whether the executable script is malware.

5. (Currently Amended) A tangible computer-readable medium bearing computer-executable instructions which, when executed on a computing device, carry out ~~the a~~ a method for determining whether a computer-executable script is malware, comprising:

obtaining an executable script;

generating a first normalized signature for the executable script, wherein the first normalized signature comprises normalized tokens ~~translated~~ normalized from corresponding functional contents of the executable script in a format suitable for comparison to normalized signatures of known malware;

comparing the first normalized signature to at least one normalized signature of known malware scripts;

determining, based on the previous comparison, whether the executable script is malware; and

reporting the results of the determination as to whether the executable script is malware.

6. (Currently Amended) The malware detection system of Claim 2, wherein ~~translating~~ normalizing tokens from the executable script into a common format suitable for comparison with the at least one malware signature in the malware signature store comprises renaming tokens from the executable script according to a common naming convention.

7. (Currently Amended) The malware detection system of Claim 6 further configured to:

if the prior determination indicates that the executable script is a partial match to at least one malware signature in the malware signature store:

generate a second normalized signature for the executable script, wherein generating a second normalized signature comprises ~~translating~~ normalizing tokens from the executable script into a second common format suitable for comparison with a second normalized malware signature of known malware in the malware signature store; and

determine whether the executable script is malware according to a comparison between the second normalized signature and at least one second normalized signature in the malware signature store.

8. (Currently Amended) The malware detection system of Claim 7, wherein ~~translating~~ normalizing tokens from the executable script into a second common format suitable for comparison with a second normalized malware signature of known malware in the malware

signature store comprises ~~translating~~ normalizing tokens of the executable script into a common name according to each token's type.

9. (Previously Presented) The malware detection system of Claim 6, wherein generating a normalized signature for the executable script further comprises generating a set of normalized tokens for each routine in the executable script.

10. (Previously Presented) The malware detection system of Claim 3, wherein determining whether the executable script is malware according to the comparison performed by the comparison means comprises determining whether the comparison found a complete match between the normalized signature for the executable script and a normalized malware signature in the malware signature store means and if so, reporting that the executable script is malware.

11. (Currently Amended) The malware detection system of Claim 10, wherein determining whether the executable script is malware according to the comparison performed by the comparison means further comprises:

determining whether the comparison found a partial match between the normalized signature for the executable script and a normalized malware signature in the malware signature store and if so:

generating a second normalized malware signature for the executable script, the second normalized signature comprising tokens from the executable script ~~translated~~ normalized into a second common format suitable for comparison with second normalized malware signatures of known malware in the malware signature store means; and

comparing the second normalized signature for the executable script to second normalized signatures of known malware in the malware signature store means to determine whether the second normalized signature for the executable script is a complete match to a second normalized signature of known malware in the malware signature store means, and if so, reporting that executable script is malware.

12. (Currently Amended) The malware detection system of Claim 11, wherein

~~translating~~ normalizing tokens from the executable script into a second common format suitable for comparison with second normalized malware signatures of known malware in the malware signature store means comprises ~~translating~~ normalizing tokens of the executable script into a common name according to each token's type.

13. (Previously Presented) The method of Claim 4, wherein determining, based on the previous comparison, whether the executable script is malware comprises determining if the first normalized signature for the executable script is a complete match with a normalized signature of known malware, and if so, reporting that the executable script is malware.

14. (Currently Amended) The method of Claim 13, wherein determining, based on the previous comparison, whether the executable script is malware further comprises:

determining if the first normalized signature for the executable script is a partial match with a normalized signature of known malware, and if so:

generating a second normalized malware signature for the executable script, the second normalized signature comprising tokens from the executable script ~~translated~~ normalized into a second common format suitable for comparison with second normalized malware signatures of known malware; and

comparing the second normalized signature for the executable script to second normalized signatures of known malware to determine whether the second normalized signature for the executable script is a complete match to a second normalized signature of known malware, and if so, reporting that the executable script is malware.

15. (Currently Amended) The method of Claim 14, wherein ~~translating~~ normalizing tokens from the executable script into a second common format suitable for comparison with second normalized malware signatures of known malware comprises ~~translating~~ normalizing tokens of the executable script into a common name according to each token's type.

16. (Previously Presented) The method of Claim 14 further comprising comparing the second normalized signature for the executable script to second normalized signatures of known malware to determine whether the second normalized signature for the executable script is a

partial match to a second normalized signature of known malware, and if so, reporting that the executable script is potential malware.

17. (Previously Presented) The computer-readable medium of Claim 5, wherein determining, based on the previous comparison, whether the executable script is malware comprises determining if the first normalized signature for the executable script is a complete match with a normalized signature of known malware, and if so, reporting that the executable script is malware.

18. (Currently Amended) The computer-readable medium of Claim 17, wherein determining, based on the previous comparison, whether the executable script is malware further comprises determining if the first normalized signature for the executable script is a partial match with a normalized signature of known malware, and if so:

generating a second normalized malware signature for the executable script, the second normalized signature comprising tokens from the executable script ~~translated~~ normalized into a second common format suitable for comparison with second normalized malware signatures of known malware; and

comparing the second normalized signature for the executable script to second normalized signatures of known malware to determine whether the second normalized signature for the executable script is a complete match to a second normalized signature of known malware, and if so, reporting that the executable script is malware.

19. (Currently Amended) The computer-readable medium of Claim 18, wherein ~~translating~~ normalizing tokens from the executable script into a second common format suitable for comparison with second normalized malware signatures of known malware comprises ~~translating~~ normalizing tokens of the executable script into a common name according to each token's type.

20. (Previously Presented) The computer-readable medium of Claim 19, wherein the method further comprises comparing the second normalized signature for the executable script to

second normalized signatures of known malware to determine whether the second normalized signature for the executable script is a partial match to a second normalized signature of known malware, and if so, reporting that the executable script is potential malware.

21. (New) A computing device configured with computer-executable instructions to perform a process to detect malware, the process including:

- parsing an executable script to obtain a plurality of tokens therefrom, the plurality of tokens including tokens respectively corresponding to variables and subroutines of the executable script;

- if a token of the plurality of tokens obtained corresponds to a variable, generating a variable token based on renaming the variable;

- if a token of the plurality of tokens obtained corresponds to a subroutine, generating a subroutine token based on renaming the subroutine;

- forming a token set from the variable token and the subroutine token;

- comparing the token set with a token set of a known malware script; and

- if there is a match, reporting that the executable script is malware.

22. (New) The computing device of claim 21, the parsing further including obtaining tokens respectively corresponding to operators, constants, execution directives, comments and white space.